

Autoevaluación

1. ¿Qué nivel de RAID emplearemos si tenemos 3 discos de 4TB cada uno y queremos poder ver una única unidad de 8TB de modo que no se pierdan datos si uno de los discos se rompe?
 - RAID 0.
 - RAID 1.
 - RAID 3.
 - **RAID 5.** RAID 5 permite usar 3 discos para obtener una capacidad de 8TB con redundancia para tolerancia a fallos
2. ¿Cómo se consigue la redundancia en los sistemas de ficheros distribuidos?
 - Gracias a que en cada nodo hay al menos 2 discos duros conectados en RAID.
 - No se consigue redundancia.
 - **Gracias a que el contenido de cada fichero es almacenado en más de un nodo del clúster.**
 - Gracias a que el contenido de cada fichero es almacenado en todos los nodos del clúster.
3. ¿Como se consigue la durabilidad de los datos en sistemas de almacenamiento distribuido en memoria?
 - **No se consigue a menos que realmente no sea sólo en memoria.**
 - No se consigue porque la memoria RAM siempre puede fallar en algún momento.
 - Se consigue gracias a que siempre queda al menos un nodo funcionando con todos los datos en memoria.
 - **Se consigue gracias a que los datos están replicados de modo que con que queden 2 nodos funcionando es suficiente para no perder nada.**
4. ¿Cuál es la especificación recomendada para los nodos que un clúster HDFS?
 - Máquinas específicas de alto precio, con fuente de alimentación y discos duros redundantes.
 - **Máquinas que puede comprar el cliente final, pero de altas prestaciones.**
 - Será suficiente con las máquinas más baratas que se puedan encontrar siempre que haya muchas.
 - Máquinas adquiridas de segunda mano, aunque sean antiguas.
5. Siento los identificadores de bloque en HDFS de 64 bits (de modo que puede indexar hasta 2^{64} bloques), piensa cuál sería el tamaño máximo de un fichero en un clúster HDFS cuyo tamaño de bloque esté configurado como 128 MB.

El tamaño máximo de un archivo sería $2^{64} \times 128 \text{ MB}$, esto da un valor astronómico: alrededor de 2.147.483.648 terabytes.
6. ¿Por qué razón los bloques en HDFS son por defecto de 128MB?
 - Porque de ese modo en muchas ocasiones todo el fichero cabe en un único bloque.
 - **Son grandes para minimizar los accesos a disco y así poder trabajar prácticamente a la velocidad de transferencia de la unidad de almacenamiento.**
 - Los bloques son de tamaños arbitrarios.
 - Los bloques no son de 128 MB sino de 128 kB.

7. ¿Cómo se puede acceder a HDFS?

- Sólo por línea de comandos.
- **Por línea de comandos o a través de lenguajes de programación usando librerías.**
- Únicamente a través de lenguajes de programación usando librerías.
- No accedemos, porque HDFS opera de modo independiente.

8. ¿Cómo se puede acceder a HDFS desde Python?

- No se puede.
- **A través de librerías.**
- El propio lenguaje lo permite por defecto.
- Sólo mediante llamadas al sistema a *hadoop fs* o a *hdfs dfs*.

9. ¿Qué significa *schema-on-read*?

- Que no es necesario ningún esquema en ningún momento.
- Que hay que establecer el esquema de las tablas en el momento de hacer la primera lectura.
- Que tanto datos como esquema tienen que coincidir cuando se hacen las escrituras.
- **Que los datos se pueden escribir sin atender a ningún esquema preestablecido, porque si se utiliza algún tipo de esquema es en todo caso al leerlos.**

10. ¿A qué nos referimos si decimos que empleamos *sharding* en un clúster?

- A que los datos se almacenan de forma distribuida.
- A que los nodos trabajan todos al mismo nivel de jerarquía.
- A que cada dato está replicado en más de un nodo.
- **A que los conjuntos de datos se han particionado en subconjuntos más pequeños para facilitar su distribución por el clúster.**

11. Una base de datos con replicación par-a-par y gestión optimista de la concurrencia ¿se puede perder información relacionada con escrituras?

Sí, si no se gestionan correctamente los conflictos.

12. ¿A qué nos referimos si decimos que empleamos replicación en un clúster?

- A que los conjuntos de datos se han particionado para facilitar su distribución por los nodos.
- **A que cada dato se almacena al menos en 2 nodos.**
- A que cada dato se almacena al menos en 3 nodos.
- A que los nodos cuentan con parejas de discos duros en RAID.

13. ¿Cuál de las siguientes afirmaciones es correcta en relación a *sharding* y replicación?

- **Ambas pueden funcionar a la vez.**
- Primero se pone a funcionar de forma replicada y después se activa el *sharding*.
- Primero se pone a funcionar con *sharding* y después se activa la replicación.
- En teoría son cosas distintas, pero en la práctica resultan intercambiables.

14. ¿Cuál de las siguientes afirmaciones es correcta en relación a las bases de datos documentales?

- Se utilizan para almacenar documentos de *Microsoft Office*.

- Se utilizan para almacenar de forma segura documentación legal de cualquier tipo.
- **Guardan documentos en formato textual, codificado típicamente en JSON.**
- Almacenan tablas cuyas celdas son documentos.

15. ¿En qué se parecen las bases de datos documentales y las que son clave-valor?

Ambas son NoSQL, y aunque las documentales son más complejas, comparten la idea de almacenar datos con flexibilidad sin esquemas rígidos.

16. ¿Cuál de las siguientes afirmaciones es correcta en relación a las bases de datos clave-valor?

- Guardan claves arbitrarias asociadas a valores predefinidos.
- Guardan claves predefinidas asociadas a valores arbitrarios.
- Guardan valores predefinidos asociados a claves arbitrarias.
- **Guardan valores arbitrarios asociados a claves.**

17. ¿Cuál de las siguientes afirmaciones es correcta en relación a las bases de datos columnares?

- **En ellas los datos se almacenan por columnas.**
- En ellas cada registro viene representado por una columna.
- En ellas una columna puede contener otras columnas.
- Son equivalentes a las relacionales salvo por una cuestión de nomenclatura.

18. ¿Cuál de las siguientes afirmaciones es correcta en relación a las bases de datos orientadas a grafo?

- En ellas cada columna es un grafo.
- En ellas cada celda es un grafo.
- En ellas cada celda de tabla se representa en modo de nodo de grafo.
- **Tanto los nodos como las relaciones entre ellos pueden contener propiedades.**

19. ¿Cómo cambiamos entre bases de datos desde el intérprete de comandos de MongoDB?

- `change_to <db>`
- **`use <db>`**
- `open <db>`
- `init <db>`

20. ¿Qué significa lo siguiente en el intérprete de comandos de MongoDB?

```
db.movies.find( { "awards.wins": { $gt: 100 } } );
```

- Dentro de la base de datos *movies*, encontrar documentos cuyo campo *wins* tenga valor 100.
- Dentro de la colección *movies* de la base de datos *db*, encontrar documentos cuyo campo *wins* tenga valor 100.
- Dentro de la colección *movies* de la base de datos *db*, encontrar documentos cuyo campo *wins* dentro del campo *awards* tenga valor 100.
- **Dentro de la colección *movies* de la base de datos *db*, encontrar documentos cuyo campo *wins* dentro del campo *awards* tenga valor mayor de 100.**